A semantic analysis of action verbs based on physical primitives

Jugal K. Kalita and Norman I. Badler
Department of Computer & Information Science
University of Pennsylvania
Philadelphia, PA 19104-6389

Abstract

We develop a representation scheme for action verbs and their modifiers based on decompositional analysis emphasizing the implementability of the underlying semantic primitives. Our primitives pertain to mechanical characteristics of the tasks denoted by the verbs; they refer to geometric constraints, kinematic and dynamic characteristics, and certain aspectual characteristics such as repetitiveness of one or more sub-actions, and definedness of termination points.

1 Introduction

Suppose a human agent is asked to perform the following commands in a suitable environment:

- *Put the block on the table.*
- *Turn the switch to position 6.*
- *Roll the ball across the table.*
- *Open the door.*

Each of these sentences specifies an underlying task requested of an agent. In order to perform the task, the performing agent has to "understand" the command. Understanding the imperatives requires understanding the meanings of the action verbs such as *put, turn, open* and *roll*, and the meanings of prepositional words such as *on, in* and *across*. One has to integrate the meanings of the constituents and produce a meaning of the sentence as a whole taking pragmatic factors into consideration, wherever appropriate. Having done so, one has to construct a plan for execution of the task in the environment. Only then the agent may perform the action. All the above steps need to be followed even if the agent is not human, but program-controlled such as an animated agent in a computer graphics environment or a robotic agent.

Among the myriad issues involved in the comprehension of imperatives in a physical domain and the execution of the underlying tasks, this paper primarily deals with developing a representation for the meanings of verbs and prepositions in order to characterize underlying actions. Following Badler [Badler 1989b], we develop a representation in which movements denoted by action verbs can be decomposed into "primitives" with implementable semantics.

2 Identifying the nature of components

Case frames [Fillmore 1980], or its variations have been used extensively for representing the semantic roles between noun phrases and the verb in a clause. Comprehensive analysis of case frames have indicated limitations of their representational capabilities [Palmer 1985, Levin 1979, Jackendoff 1972]. These include the existence of a large number of exceptions, lack of consensus in establishing a universal set of cases, invalidity of multiple case assignments to a noun phrase, unsubstantiated semantic overloading of the names of cases, lack of explicit characterization of inter-relationships among cases, and inadequacy of cases for representing the full complexity of concepts denoted by verbs. Considering the fact that a decompositional analysis obviates most of the above drawbacks [Palmer 1985], we opt for such an approach.

Palmer's research [Palmer 1985] in the domain of word problems in physics is similar to ours in objective and approach. However, our approach is "active", laying emphasis on implementability of the primitives, which is not the case in her study. Although her study considers some simple motion verbs, she treats them in a "static" fashion. Miller's analysis [Miller 1972] provided English paraphrases of some complex motion verbs in terms of "simpler" ones; Okada's analysis [Okada 1980] lacks in coherence and proper justifications. Schank's representation of verbs [Schank 1972], though impressive, suffers from the lack of uniformity in the levels of decomposition and explicit semantic anchoring of the primitives in terms of executability at a non-linguistic level.

Componential analysis performed here involves working from bottom up rather than from the language end. Our goal is to obtain representations as close to the physical world as possible. We obtain the components of the verbs from an analysis of the "real" actions they represent taking into consideration physical attributes only. Thus, we use the linguistic technique of componential analysis for the representation of verbal meanings, but confine ourselves to consideration of mechanistic components only. This view of verbal semantics, albeit incomplete, is not short-sighted; such a study has not been carried out in an elaborate fashion to date and is vitally essential for understanding the physical nature of actions.

The attributes we identify for the characterization of the nature of tasks denoted by action verbs deal with the following criteria: kinematic/dynamic distinctions; the type and operational nature of relevant geometric constraints, aspectual considerations such as repetitiveness, whether termination condition(s) for the verb's underlying task are naturally well-defined, or need to be inferred contextually, etc. In prevalent characterizations of actions, kinematic and dynamic characteristics of verbs have not been considered in detail. Badler mentions these attributes, but does not elaborate [Badler 1989b].

2.1 Geometric constraints

Geometric constraints can be used to clearly describe certain movements and configurations of physical objects. They provide information regarding how one or more objects or sub-parts of objects relate to one another in terms of physical contact, absolute or relative location, inter-object distance, absolute and relative orientation, or path of motion. Nelson [Nelson 1985], Rossignac [Rossignac 1986] and Barzel [Barzel and Barr 1988] use geometric constraints to describe the motion of "simple" objects. Badler discusses specification of motions through constraints for succinct expression of activities of articulated human figures composed of hierarchic sub-parts performing natural actions or tasks [Badler 1989b, Badler 1989a].

2.1.1 Constraint types

Some verbs' underlying actions can be primarily described by positional or orientational constraints.

1. *Positional constraints*: This refers to situations in which a 0-, 1-, 2- or 3-dimensional object is constrained to a 0-, 1-, 2- or 3-dimensional region of space. For example, in order to execute the command *Put the ball on the table*, an arbitrary point on the surface of the ball has to be brought in contact with (or constrained to) an arbitrary point on the surface of the table. Another example is seen in the action underlying the imperative *Put the block in the box* where one needs to constrain the block (or the volume occupied by the block) to the interior volume of the box.
2. *Orientational constraints*: Consider the meaning of the preposition *across* in the sentence *Place the ruler across the table*. The interpretation of the preposition involves several components, one of which requires that the longitudinal axis of the ruler and the longitudinal axis of the table top be perpendicular to each other. This requirement can be expressed in terms of an orientational constraint.

2.1.2 Constraint operation

Verbs dealing with constraints can be classified considering whether they denote establishment, removal, maintenance or modification of (existing) geometric constraints.

1. Verbs whose central action requires that constraints established continue to hold: attach, hold, engage, fix, grab, grasp, hook.
2. Verbs whose central action requires that already existing constraints *cease* to hold. Examples include: detach, disconnect, disengage, release.
3. Verbs which refer to modification of an already existing constraint: loosen, tighten. We do not discuss such verbs in this paper.

2.2 Aspectual components

Aspect is an inherent semantic content of a lexical item pertaining to the temporal structure of the situation denoted by the lexical item, independent of context [Passonneau 1988]. Nakhimovsky's notion of aspectual class refers to internal temporal structuring of generic situations [Nakhimovsky 1988]. Moens's use of the term aspectual category is different in that it takes speaker's perspectives into consideration [Moens and Steedman 1988]. Below, we consider two aspectual characteristics: repetitiveness and telicity.

2.2.1 Repetitiveness or frequentation

1. Verbs whose underlying tasks definitely need repetitions of one or more sub-actions: roll, calibrate, screw, scrub, shake, rock.
2. Verbs for which repetitions may or may not be performed (i.e., whether repetitions are performed depends on the object(s) involved, information gathered from linguistic input, etc). Some example verbs are: cut, fill, lace, load, tamp.

2.2.2 Terminal conditions

We can divide the tasks identified by verbs into two categories based on this criterion. Nakhimovsky [Nakhimovsky 1988] also discusses the telicity property of actions denoted by verbs and verbal phrases.

1. **Atelic verbs**: The tasks denoted by such verbs do not have properly defined end conditions. The termination point may be determined by accompanying linguistic expressions or by context-dependent, task-dependent criteria such as default resulting states of affected object(s), or obtained through reasoning from commonsense knowledge, or knowledge of the goal to achieve, or defined by explicit feedback from simulation. For example, *cut* is atelic because an object can be cut and recut repeatedly, unless the linguistic imperative specifies that the object be cut once, or twice, etc., or specifies the size of the resultant pieces. Other examples of such verbs are: fold, hold, press, scrub, shake.
2. **Telic verbs**: These are verbs whose underlying tasks have properly defined built-in terminal points that are reached in the normal course of events and beyond which the processes cannot continue. Some examples are: align, assemble, attach, close, detach, drop, engage, fix, fill, place, release.

2.3 Kinematic—dynamic characterization of actions

Badler [Badler 1989b] states that an approach to movement (action) representation should be able to characterize (qualitative) specifications of kinematic and dynamic information, whenever appropriate. Dynamics describes the force or effort influencing motion. Kinematics deals with direct path or goals, and motion specification. Often movements along the same spatial path and toward the same spatial goal may be represented by different verbs such as *touch*, *press* and *punch*. These distinctions can be formulated in terms of dynamic specification.

1. *Kinematic:* These are verbs whose underlying actions can be expressed as a movement along an arbitrary path or at an arbitrary velocity. Some examples are turn, roll, rotate.
2. *Dynamic:* For a verb in this category, its underlying action can be characterized by describing the force which causes it. Examples include: push, shove, pull, drag, wring, hit, strike, punch, press.
3. Both kinematic and dynamic: These are verbs which have strong path as well as force components: swing, grip/grasp (vs. touch), twist.

3 Obtaining a representation for the verbs: primitives

The primitives we use are anchored outside the linguistic domain. Currently, we work in the domain of graphical animation and hence, our primitives are executable directly in a graphics environment called JACK [Phillips 1988]. The expression of linguistic semantics in terms of tested primitives vindicates the usefulness and "completeness" of our approach. Implementation of the dynamic primitives used can be found in [Otani 1989]. They pertain to forces, torques, mass and balance. We do not discuss dynamic primitives here since they are not directly relevant to the examples in this paper. Implementation of the primitives concerned with geometric relations are discussed in [Zhao and Badler 1989].

3.1 Geometric relations and geometric constraints

We specify geometric relations in terms of the following frame structure.

```
Geometric-relation:  spatial-type:
                     source-constraint-space:
                     destination-constraint-space:
                     selectional-restrictions:
```

Spatial-type refers to the type of the geometric relation specified. It may have one of two values: ***positional*** and ***orientational***. The two slots called ***source-constraint-space*** and ***destination-constraint-space*** refer to one or more objects, or parts or features thereof which need to be related. For example, in order to execute the command ***Put the cup on the table***, one brings the bottom surface of the cup into contact with the top surface of the table. The command ***Put the ball on the table*** requires bringing an arbitrary point on the surface of the ball in contact with the surface of the table top. Since, the items being related may be arbitrary geometric entities (i.e., points, surfaces, volumes, etc.), we call them *spaces*; the first space is called the source space and the second the destination space. The slot *selectional-restrictions* refers to conditions (static, dynamic, global or object-specific) that need to be satisfied before the constraint can be executed.

Geometric constraints are geometric goals; they are specified as follows:

Geometric-constraint: execution-type:
geometric-relation:

Geometric constraints are of four types. They are distinguished by the ***execution-type*** component. The execution class or type of a constraint may be *achieve, break, maintain* or *modify*.

3.2 Kinematics

The frame used for specifying the kinematic aspect of motion is the following:

Kinematics: motion-type:
source:
destination:
path-geometry:
velocity:
axis:

Motions are mainly of two types: *translational* and *rotational*. In order to describe a translational motion, we need to specify the source of the motion, its destination, the trajectory of the path, and the velocity of the motion. In the case of rotational motion, the path-geometry is always circular. The velocity, if specified is angular. An axis of rotation should be specified; otherwise, it is inferred by consulting geometric knowledge about the object concerned.

3.3 Kernel actions

The central part of an action consists of one or more components: ***dynamics, kinematics*** and ***geometric-constraints***— along with control structures stating aspectual or other complexities involved in the execution of an action. The constructs we use in the paper are: *repeat-arbitrary-times* and *concurrent*. The keyword *concurrent* is specified when two or more components, be they kinematic, dynamic or geometric constraints, need to be satisfied or achieved at the same time. The keyword *repeat-arbitrary-times* provides a means for specifying the frequentation propery of certain verbs. The verbs' semantic representation need not specify how many times the action or sub-action may need to be repeated. However, since every action is presumed to end, the number of repetitions of an action will have to be computed from simulation (based on tests for some suitable termination conditions), or by inference unless specified linguistically as in *Shake the block about fifty times*.

3.4 Representation of verbal and sentential meaning

Since our meaning representation is verb-based, the template for the representation of the meaning of a verb is also the frame for representation of meanings of sentences. The representation for a sentence has the following slots.

Verbal-representation: agent:
object:
kernel-actions:
selectional-restrictions:

Selectional restrictions may refer to dynamic or static properties of objects or the environment.

4 Some example verbs

4.1 A verb of establishment of positional constraint: *put*

Webster's dictionary [Woolfe 1981] defines one sense of the meaning of the verb *put* as *to place in a specified position or relationship*. We consider only the positional aspect of the meaning to obtain a lexical definition. The nature of the positional constraint is dependent upon the nature of the object, and the location specified by a locative expression. The lexical entry is

```
put (l-agent, l-object, l-locative) ⟵ agent: l-agent
                                      object: l-object
                                      kernel-action:
                                           geometric-constraint:
                                                  execution-type:achieve
                                                  spatial-type:   positional
                                                  geometric-relation: l-locative
```

This representation tells us that *put* requires us to *achieve* a *positional* constraint between two objects, parts or features thereof. It does not indicate the type of positional relation to be achieved. The details of the geometric relation to be achieved have to be provided by the locative expression used which may be in terms of prepositions such as *in, on* or *across*.

4.2 A kinematic verb: *roll*

The verb *roll* refers to two motions occurring concurrently: a rotational motion about the longitudinal axis of the object and a translational motion of the object along an arbitrary path. The rotational motion is repeated an arbitrary number of times. The representation for the verb *roll* we use is as follows:

```
roll (l-agent, l-object, path-relation)⟵
        agent: l-agent
        object:l-object
        kernel-action:
             concurrent { { kinematic:
                                  motion-type:  rotational
                                  axis:         longitudinal-axis-of (l-object)
                          } repeat-arbitrary-times }
                          { kinematic:
                                  motion-type:  translational
                                . path:         path-relation } }
        Selectional Restrictions:  has-circular-contour (l-object, longitudinal-axis-of (l-object))
```

4.3 A verb that removes constraints: *open*

We consider just one sense of *open*—the sense defined by Webster's Dictionary [Woolfe 1981] as *to move (as a door) from closed position*. The meaning is defined with respect to a specific position of the object under consideration. The closed position of the object can be viewed as a *constraint* on the position or orientation of the object. Thus, *open* can be considered as a verb whose underlying task undoes an existing *constraint*. The object under consideration is required to have at least two parts: a solid 2-dimensional part called the *cover* and an unfilled 2-dimensional part defined by some kind of frame: the *hole*. The meaning must capture that the agent performs an action whose result is to remove the constraint that object's cover and its hole are in one coincident plane. Additionally, the object's cover must occupy the total space available in object's hole in the constrained position. This is fulfilled by requiring that the two (sub-)objects (the hole and the cover) are of the same shape and size.

The definition for *open* is:

```
open (Ag, Obj) ⟵ agent: Ag
                 object: Obj
                 kernel-action:
                      geometric-constraint:
                             execution-type:break
```

spatial-type: positional
geometric-relation: source-constraint-space: Obj • hole
destination-constraint-space: Obj • cover
Selectional Restrictions: contains-part (Obj, hole)
contains-part (Obj, cover)
area-of (Obj.cover) = area-of (Obj • hole)
shape-of (Obj.cover) = shape-of (Obj • hole)

5 Representing meanings of prepositions

In order to provide precise, implementable meanings of prepositions, we have been influenced by Badler [Badler 1975] and Gangel [Gangel 1984]. The semantics of the linguistic locative expression has also been discussed by Herskovits [Herskovits 1986]. Talmy's work [Talmy 1983] on the relation between language and space also discusses semantic representation of prepositions. We obtain lexical entries for a selection of prepositions in [Kalita 1990]. We include one such example below. Our definitions are limited in that they work with simple, solid, non-deformable geometric objects.

5.1 A representation for *on*

Among the senses of *on* defined by Herskovits [Herskovits 1986], the one we are interested in is: *spatial entity supported by physical object.* Examples of its use are seen in sentences such as *Put the block on the table* and *Put the block on the box.* The support can come in various forms as discussed by Herskovits. One situation which is commonplace or prototypical, is the one in which the located object rests on a free, horizontal, upward facing surface of the reference object; this need not be a top surface of the reference object, though it almost always is an *outer* surface (otherwise *in* is preferred). Webster's dictionary [Woolfe 1981] defines this meaning as *a function word to indicate a position over and in contact with.* We describe this meaning of *on* as

on (X,Y) ⟵ geometric-relation:
spatial-type: positional
source-constraint-space: any-of (self-supporting-spaces-of (X))
destination-constraint-space: any-of (supporter-surfaces-of (Y)))
selectional-restrictions:
horizontal (destination-constraint-space)
equal ((direction-of (normal-to destination-constraint-space) "global-up")
area-of (source-constraint-space) $\leq$ area-of (destination-constraint-space)
free-p (destination-constraint-space)

Given a geometric object, the geometrical function *self-supporting-spaces-of* obtains a list containing surfaces, lines or points on the object on which it can support itself. For example, a cube can support itself on any of its six faces, and a sphere on any point on its surface. The function *supporting-surfaces-of* finds out the surfaces on an object on which other objects can be supported. The functions *direction-of, normal-to, horizontal-p* and *area-of* are self-evident. The two directional constants *global-up* and *global-down* are defined with respect to a global reference frame.

6 Processing the sentence *Put the block on the table*

The sentence consists of the action verb *put.* The object is specified as the subject of the sentence and the location as a prepositional phrase. The meaning of the whole sentence, obtained by composing the meanings of its constituent parts [Kalita 1990] as specified by the definitions for *put* and *on*, is

agent: "you"
object: block-1
kernel-action: geometric-constraint:
execution-type: achieve
spatial-type: positional
geometric-relation:
spatial-type: positional
source-constraint-space: any-of (self-supporting-spaces-of (block-1))

destination-constraint-space: any-of (supporter-spaces-of (table-1)))
selectional-restrictions:
horizontal-p (destination-constraint-space)
equal (direction-of (normal-to destination-constraint-space) "global-up")
area-of (source-constraint-space) $\leq$ area-of (destination-constraint-space)
free-p (destination-constraint-space)

In order to execute the action dictated by this sentence, the program [Kalita 1990] looks at the knowledge stored about the block to find a part or feature of the block on which it can support itself. It can be supported on any one of its faces and no face is more salient than any other for supporting purposes. A cube (the shape of the block) has six faces and one is chosen randomly as the support area. The program searches the knowledge stored about the table for a part or feature which can be used to support other objects. It gathers that the table's function is to support "small" objects on its top which is also horizontal as required by a selectional restriction. Finally, the system concludes that one of the sides of the cube has to be brought in contact with the top of the table. Herskovits [Herskovits 1986] gives a very general discussion on the properties or attributes of objects reqired for such inferences. A planning module is needed to make the interface between the semantic representation level and the animation level complete. Details of object knowledge representation and planning can be found in [Kalita 1990]. Finally, the simulation program YAPS [Esakov and Badler 1990] performs appropriate computations and inferences where necessary, taking into consideration many aspects such as detailed geometric knowledge, default states, and knowledge of strengths and dimensions of the various parts of the body of the agent, and perform a graphical animation of the task's execution.

7 Conclusions

We have demonstrated that operational meanings of action verbs and their modifiers can be represented in terms of components pertaining to aspectuals, constraints, and kinematic/dynamic characterization. For additional examples of decomposition, the reader is referred to [Kalita 1990].

We have implemented a system incorporating the semantic processing discussed in this paper in Common Lisp on a HP workstation. The semantic output is further processed in consultation with the detailed knowledge stored about the objects under consideration. Finally, this interpreted output is used by YAPS [Esakov et al 1989, Esakov and Badler 1990] to drive a graphical animation. The successful animation of tasks starting from natural language input provides a sound anchoring for our semantic representation.

In this paper, we have not discussed the classification of lingusitic arguments of a verb as obligatory or optional, and the treatment of different classes of arguments. This issue is discussed in [Kalita 1990]. We have not also considered significant aspects of language usage such as intention, non-geometric goals and beliefs of agents. The reason is that ours is an in-depth study of the purely physical aspects of actions; thus, direct physical realizability is of primary importance. Our study has to be complemented by augmenting it with such intensional considerations for a fuller understanding.

Acknowledgements

This research is partially supported by Lockheed Engineering and Management Services (NASA Johnson Space Center), NASA Ames Grant NAG-2-426, FMC Corporation, Martin-Marietta Denver Aerospace, NSF CER Grant MCS-82-19196, and ARO Grant DAAL03-89-C-0031 including participation by the U.S. Army Human Engineering Laboratory.

References

[Badler 1975] Badler, Norman I., 1975. *Temporal Scene Analysis: Conceptual Description of Object Movements.* Technical Report TR-80, University of Toronto, Toronto, Ontario. (University of Pennsylvania, Department of Computer and Information Science, Technical Report 76-4).

[Badler 1989a] Badler, N.I., 1989. Artificial Intelligence, Natural Language and Simulation for Human Animation. In Magnenat-Thalmann, M. and Thalmann, D. (editors), *State-of-the-Art in Computer Animation.* Springer Verlag, New York.

[Badler 1989b] Badler, N.I., 1989. A Representation for Natural Human Movement. In Gray, J.A. (editor), *Dance Technology: Current Applications and Future Trends.* AAHPERD Publications, Reston, VA.

[Barzel and Barr 1988] Barzel, R. and Barr, A.H., 1988. A Modeling System Based on Dynamic Constraints. *Computer Graphics* 22(4):179–188.

[Esakov and Badler 1990] Esakov, Jeffrey and Badler, Norman I., 1990. An Architecture for High-Level Human Task Animation Control. In Fishwick, P.A. and Modjeski, R.S. (editors), *Knowledge-Based Simulation: Methodology and Aplications*. Springer Verlag, New York.

[Esakov et al 1989] Esakov, J.; Jung, M.; and Badler, N. I., 1989. An investigation of language input and performance timing for task animation. In *Proceedings of Graphics Interface 89*, pages 86–93. Morgan-Kaufmann, Palo Alto, Ca.

[Fillmore 1980] Fillmore, C., 1980. The Case for Case. In Bach and Harms (editors), *Universals in Linguistic Theory*, pages 1–88. Holt, Reinhart and Winston, New York.

[Gangel 1984] Gangel, Jeffrey, 1984. A Motion Verb Interface to a Task Animation System. Master's thesis, Department of Computer and Information Science, University of Pennsylvania.

[Herskovits 1986] Herskovits, Annette, 1986. *Language and Spatial Cognition. Studies in Natural Language Processing*. Cambridge University Press, Cambridge, England.

[Jackendoff 1972] Jackendoff, R.S., 1972. *Semantic Interpretation in Generative Grammar*. MIT Press, Cambridge, Massachussetts.

[Kalita 1990] Kalita, J.K., 1990. Analysis of Some Actions Verbs and Synthesis of Underlying Tasks in an Animation Environment. Forthcoming Ph.D. Thesis, Department of Computer and Information Science, University of Pennsylvania.

[Levin 1979] Levin, B., 1979. Predicate-Argument Structures in English. Master's thesis, MIT.

[Miller 1972] Miller, G.A., 1972. English Verbs of Motion: A Case Study in Semantics and Lexical Memory. In Melton, A. W. and Martin, E. (editors), *Coding Processes in Human Memory*, pages 335–372. V.H. Winston and Sons.

[Moens and Steedman 1988] Moens, Marc and Steedman, Mark, June 1988. Temporal Ontology and Temporal Reference. *Computational Linguistics* 14(2):15–28.

[Nakhimovsky 1988] Nakhimovsky, A., June 1988. Aspect, Aspectual Class and the Temporal Structure of Narrative. *Computational Linguistics* 14(2):29–43.

[Nelson 1985] Nelson, G., 1985. Juno, a Constraint-based Graphics System. *Computer Graphics* 19(3):235–243.

[Okada 1980] Okada, N., 1980. A Conceptual Taxonomy of Japanese Verbs for Understanding Natural Language and Picture Patterns. In *Proceedings of ICCL*, pages 127–135.

[Otani 1989] Otani, Ernest, 1989. Software Tools for Dynamic and Kinematic Modeling of Human Motion. Master's thesis, Department of Mechanical Engineering, University of Pennsylvania, Technical Report, MS-CIS-89-43, Philadephia, PA.

[Palmer 1985] Palmer, Martha, 1985. *Driving Semantics in a Limited Domain*. PhD thesis, University of Edinburgh, Scotland.

[Passonneau 1988] Passonneau, R.J., June 1988. A Computational Model of the Semantics of Tense and Aspect. *Computational Linguistics* 14(2):44–60.

[Phillips 1988] Phillips, C.B., 1988. *JACK User's Guide*. Computer Graphics Research Laboratory, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA.

[Rossignac 1986] Rossignac, J.R., 1986. Constraints in Constructive Solid Geometry. In *Proceedings of Workshop on Interactive 3-D Graphics*, pages 93–110. ACM.

[Schank 1972] Schank, Roger, 1972. Conceptual Dependency: A Theory of Natural Language Understanding. *Cognitive Psychology* 3(4).

[Talmy 1983] Talmy, Leonard, 1983. How Language Structures Space. In Pick, H. and Acredols, L. (editors), *Spatial Orientation: Theory, Research and Application*, pages 225–282. Plenum Press, New York.

[Woolfe 1981] Woolfe, Henry (editor), 1981. *Webster's New Collegiate Dictionary*. G. & C. Merriam Company, Springfield, MA.

[Zhao and Badler 1989] Zhao, J. and Badler, N.I., 1989. *Ral Time Inverse Kinematics with Joint Limits and Spatial Constraints*. Technical Report MS-CIS-89-09, Department of Computer and Information Science, University of Pennsylvania.